IS 11289 (1985): Code of practice for performance monitoring of computer based systems [LITD 14: Software and System Engineering]

BLANK PAGE

“पुनर्पुष्ट १९९५”
"RE-AFFIRMED 1995"

IS : 11289 - 1985

(Reaffirmed 2004)

“पुनर्पुष्ट १९९०”
"REAFFIRMED 1990"

Indian Standard

CODE OF PRACTICE FOR PERFORMANCE MONITORING OF COMPUTER BASED SYSTEMS

UDC 681·31·02 : 620·16 : 006·76

INDIAN STANDARDS INSTITUTION
MANAK BHAVAN, 9 BAHADUR SHAH ZAFAR MARG
NEW DELHI 110002

Gr 7

December 1985

Indian Standard

CODE OF PRACTICE FOR PERFORMANCE MONITORING OF COMPUTER BASED SYSTEMS

Computer, Business Machines and Calculators Sectional Committee, LTDC 24

Chairman	*Representing*
Dr N. Seshagiri	Department of Electronics, New Delhi
Members	
Shri R. P. Ahuja	Computer Maintenance Corporation Ltd, New Delhi
Shri C. K. Bapiraju	State Bank of India, Bombay
Shri S. K. Bhatia	Bharat Heavy Electricals Ltd, New Delhi
Shri P. V. C. Chellapatti Rao (*Alternate*)	
Dr Vijay P. Bhatkar	Kerala State Electronics Development Corporation Ltd, Trivandrum
Dr S. N. S. Rajasekran (*Alternate*)	
Dr C. R. Chakravarthy	Ministry of Defence (R & D)
Shri K. N. Dheer	Indian Airlines, New Delhi
Shri S. P. Gothoskar	Reserve Bank of India, Bombay
Shri Deepak Gupta	Tata Electric Companies, Bombay
Shri R. M. Nair (*Alternate*)	
Dr Mathai Joseph	NCSDGT, Tata Institute for Fundamental Research, Bombay
Dr S. Ramani (*Alternate*)	
Dr A. Lahiri	Department of Science & Technology, New Delhi
Shri N. Lakshmanan	Life Insurance Corporation of India, Bombay
Shri Arjun Malhotra	Hindustan Computers Ltd, New Delhi
Shri Arun Mehta (*Alternate*)	
Dr S. C. Mehta	Steel Authority of India, New Delhi
Shri S. L. N. Murthy	Bharat Electronics Ltd, Bangalore
Shri K. S. Perinanayagam (*Alternate*)	
Shri S. K. Pandey	International Computers Indian Manufactures Ltd, Pune
Shri H. Das (*Alternate*)	
Shri V. K. R. Prabhu	Development Commissioner (Small Scale Industries), New Delhi
Shri M. Ramakrishnan (*Alternate*)	
Shri G. Raghukumar	Delhi Cloth & General Mills Co Ltd, New Delhi
Shri A.N . Pawar (*Alternate*)	

(*Continued on page* 2)

(*Continued from page* 1)

Members	*Representing*
DR V. K. RAVINDRAN	PSI Data Systems Pvt Ltd, Bangalore
SHRI V. L. DESHPANDE (*Alternate*)	
PROF R. SADANANDA	Computer Society of India, Bombay
SHRI ASHIS SEN	Indian Statistical Institute, Calcutta
SHRI UMESH P. SHAH	ORG Systems, Vadodara
SHRI P. K. SRIDHARAN (*Alternate*)	
SHRI M. SHANKRALINGAM	Directorate General of Supplies & Disposals (Inspection Wing), New Delhi
SHRI K. L. GARG (*Alternate*)	
SHRI C. G. SUBRAMANYAN	Electronics Trade and Technology Development Corporation Ltd, New Delhi
SHRI ISHWAR DUTT (*Alternate*)	
SHRI T. N. SWAMY	Electronics Corporation of India Ltd, Hyderabad
SHRI V. R. UNNIRAMAN	Telecommunication Research Centre, New Delhi
SHRI K.M. VISWANATHAN	Hindustan Teleprinter Ltd, Madras
SHRI S. DEVARAJAN (*Alternate*)	
SHRI N. SRINIVASAN, Director (Electronics)	Director General, ISI (*Ex-officio Member*)

Secretary

SHRI A. S. RAWAT
Deputy Director (Electronics), ISI

Information Systems and Procedures Subcommittee, LTDC 24 : 3

SHRI C. K. BAPIRAJU	State Bank of India, Bombay
SHRI S. K. BHATIA	Bharat Heavy Electrical Ltd, New Delhi
SHRI H. P. SAXENA (*Alternate*)	
SHRI S. P. GOTHOSKAR	Reserve Bank of India, Bombay
SHRI M. M. N. KAPUR	Central Statistical Organization, New Delhi
DR R. SHANKAR	Indian Institute of Technology, Kanpur
SHRI R. THIAGARAJAN	Department of Science and Technology, New Delhi
DR N. VIJAYADITYA	National Information Centre, New Delhi

Indian Standard

CODE OF PRACTICE FOR PERFORMANCE MONITORING OF COMPUTER BASED SYSTEMS

0. FOREWORD

0.1 This Indian Standard was adopted by the Indian Standards Institution on 29 April 1985, after the draft finalized by the Computer, Business Machines and Calculators Sectional Committee had been approved by the Electronics and Telecommunication Division Council.

0.2 This code of practice applies to all computer-based systems and covers following aspects:

a) Gives guidance in carrying out a detailed examination of the system that is to be monitored. It identifies key aspects of the system, its usage and other systems that share the same operating environment. Guidelines and check lists are provided for each of these aspects to assist the user in identifying elements of change that can affect performance;

b) Identifies a variety of monitoring techniques that may be used to measure these changes in performance; and

c) Gives guidance on the documentation of performance monitoring information.

0.3 This code gives guidance to enable organizations to:

a) assure the continuing quality of a computer-based system in service,

b) develop and apply appropriate monitoring techniques,

c) obtain early diagnosis of changes in the behaviour of the system, and

d) detect faults so that action can be taken to avoid fault propagation.

0.4 This code assumes that the system:

a) was adequately tested and documented in accordance with relevant standards,

b) performed as required under normal operating conditions, and

c) was initially schedulable within the constraints imposed by other systems and manufacturer's design.

0.4.1 This code also assumes that, in service:

a) the system runs under effective configuration management, and

b) documentation is maintained.

0.5 This standard is based on BS : 6238 : 1982 'Code of practice for performance monitoring of computer based systems', issued by the British Standards Institution.

---

1. SCOPE

1.1 This code of practice provides guidance for carrying out examination of the system with a view to monitoring and documenting its performance.

2. TERMINOLOGY

2.0 For the purposes of this standard the terms and definitions as given in IS : 1885 (Part 52) Series* shall apply.

3. KEY ASPECTS OF THE SYSTEM BEING MONITORED

3.0 The system monitoring requires a detailed examination as under for all likely causes of a change in performance and the ways in which changes to other systems sharing the same resources that may affect the performance of the system being monitored.

3.1 Applications Software

3.1.0 *General* — Some elements of change may have been anticipated during the design of the system but serious effects can arise from changes that occur, which were not visualized in the original specification.

3.1.1 *Elements of Change* — Among the elements of change that should be examined, the user should note that:

a) standards of accuracy can affect processing times,

b) security controls can extend input and output duration,

c) validation checks can extend input and output duration,

d) changes in coding of data can affect input and output times and storage capacity required,

*Electrotechnical vocabulary: Part 52 Data processing.

e) changes in code structures can affect input and output times and storage capacity required,

f) changes in the structure of the data base and files can affect input and output times and storage capacity required,

g) changes in input method/documents can increase input times,

h) changes in output method/documents can increase output times,

j) changes in parameter values can change processing times,

k) changes to algorithms or formulae could change processing times,

m) some changes may not be reflected in the procedures, and

n) some changes may not be reflected in the documentation.

3.2 Operating and Support Software

3.2.0 *General* — Changes to the operating and support software and system can have a considerable influence on the efficiency and behaviour of a system. Such changes are quite usual for various reasons (for example, correction of faults or the improvement of facilities) but a careful examination of their implications should be made to ensure that no harmful side effects will occur.

3.2.1 *Elements of Change*

3.2.1.1 *Operating software* — Operating software includes all the system facilities used in the course of running application programmes. The main consideration is that an alteration to a system facility may affect some or all application programmes, or modules, and it may not be practicable or desirable to cross-reference facilities with applications. Hence changes cannot necessarily be guaranteed to have no side effects and each one should be treated on its merits.

The main system facilities that come under this consideration are:

a) system library, holdling commonly used modules;

b) file handling system (changes here almost certainly affect all application programmes);

c) job control language (new or added commands or formats);

d) input/output; and

e) scheduling. (Changes to the priority strategy will affect run-times, elapsed times or even what time of day or night an application will be run. These in turn will affect the run-time costs.)

3.2.1.2 *Support software*:

a) *General* — This software includes all the system facilities used in the course of programme preparation. The undesirable result of changes in this area is the necessity for programmer retraining and the possible reduction in software quality and throughput until the changes are fully understood.

b) *Language translator (compilers)* — Special consideration should be given to this facility. Changes in this area may have a very significant effect on application systems performance and should be carefully vetted as they may not always be apparent. Generally, changes will be manifested as version updates or the same language from a different supplier (probably giving faster run-time code, or better store utilization, or just improved facilities). Even if a system has been extensively acceptance tested, there remains a need to understand that is inherently impossible to obtain 100 percent assurance that errors have been eliminated or all implications fully appreciated. Some of the more obvious features that may cause difficulty on an otherwise identical translator are syntax, semantics, reserved words and keywords, translator directives and linking.

Some of the less obvious features that should be considered are as follows:

1) *Arithmetic* — Although these problems should be resolved during acceptance testing of the new translator, explicit checks should be made on the following points to ensure conformity with previous software:

   i) Turncation and rounding,
   ii) The result and remainder from the division algorithm,
   iii) Equality tests for floating or mixed arithmetic, and
   iv) Overflow limits should not be less than previous limits.

2) *Expressions* — This feature is a similar problem to (a) and the following points should be checked:

   i) Order of evaluation should be identical,
   ii) Mixed 'type' conversion should give identical results, and
   iii) Intermediate state limits should not be less than previous limits.

3) *Procedures and functions* — The following points should be checked:

   i) Order of parameter evaluation should be the same; and

   ii) The parameter mechanism should behave in the same way for the same cases (by reference, value, etc). This behaviour is particularly important for default cases.

4) *Optimization* — A different optimization can give different results for some application programmes.

5) *Error handling* — Changes can cause some temporary misinterpretation.

c) *Other support software* — This category includes facilities such as:

   1) time sharing command structure,
   2) debugging tools,
   3) editor, and
   4) linker.

Changes to the linker can also interact with the system library, which is considered under operating software (*see* **3.2.1.1**).

3.2.1.3 *Software reconfiguration* — Although there may be no changes to the products that form the operating and supporting software, they may be reconfigured in such a way as to affect the performance of a system. Some of the aspects that need to be considered in these circumstances are:

a) mapping of software on to different types and speeds of media/devices,
b) minimizing the possible overheads of paging,
c) change in programme loading format and method,
d) regeneration of system software,
e) selection of new running modes such as choice of long error messages or full listings,
f) change in normal residence and running versions of software,
g) change in use of file/data catalogue system,
h) change in use of security feature and checks, and
j) change in use of application programme diagnostics.

3.3 Hardware Configuration

3.3.0 *General* — The performance of a computer-based system can be limited by one or other hardware resource. A change to the configuration can remove or introduce any such limitation or substitute one

limitation for another. The effect of configuration changes should be considered under the following heading:

a) Processor (CPU) capacity,

b) Storage devices, and

c) Input/output devices.

3.3.1 *Elements of Change*

3.3.1.1 *Processor (CPU) capacity* — A change in the speed or capacity of the processor unit can put pressure on the ability of storage input or output or any combination of these devices to cope with the resultant changes in volume of data or speed of data flow.

3.3.1.2 *Storage devices* — A balance should be maintained between the amount of main storage and the number of peripherals attached. Increasing the amount of main storage can allow more programmes to run concurrently, but will increase the loading on existing peripherals and degrade the performance of existing systems accordingly. Conversely, increasing auxiliary storage can also reduce the performance of a computer-based system.

3.3.1.3 *Input/output devices* — In the context of this code, input/output devices include automatic control devices, VDUs and sensors, etc.

Increasing the speed, number or function of these devices can affect the performance of a computer-based system. In particular, increasing the number or changing the usage mode of VDUs can increase the communications overheads with a consequent impact on other systems areas.

Greater peripheral capacity can allow more programmes to run concurrently, increasing CPU load, channel load and operating system overhead.

3.4 Hardware Ageing

3.4.0 *General* — Ageing can be a serious problem with those components of a computer configuration where motion and hence wear can occur. The two main areas that should be considered are:

a) electromechanical devices, and

b) magnetic media.

3.4.1 *Elements of Change*

3.4.1.1 *Electromechanical devices* — These devices become less reliable with age, decreasing the mean time between failures, and hence

increasing the elapsed time for jobs. Despite preventive maintenance, equipment reliability can eventually drop below the acceptance level demanded by the response-time of the system.

3.4.1.2 *Magnetic media* — Tapes and disks are subject to deterioration in use. This deterioration manifests itself in an increase in the number of read/write errors that occur. The number of hidden re-tries by the operating system can greatly increase the run-time of the programme. Excessive use of a short length of a full tape reel increases wear and should be avoided.

3.5 System Inputs

3.5.0 *General* — All computer systems should check the validity and integrity of the input data. However, there are many changes that can be made to the input data that may not be detected, and such changed data may have a wide range of adverse effects on the performance of the system.

The input data should be considered under the following headings:

a) Manual data collection systems,

b) Data preparation procedures,

c) On-line data entry,

d) Data validation,

e) Parameter data,

f) Data brought forward,

g) Real time data, and

h) Transmission channels.

3.5.1 *Elements of Change*

3.5.1.1 *Manual data collection* — Most commercial batch processing systems rely heavily on some form of manual data collection. The data collection can range from filling in forms or marks — sensing cards to collecting magnetic encoded strips or pre-punched tags.

The system performance can be affected by the following factor amongst other things on this stage:

a) New or temporary staff who have not been adequately trained,

b) Instruction manuals not kept up-to-date,

c) A new or modified form design not properly understood by one or more of the staff,

d) A new or modified means of expressing the data not properly understood by one or more of the staff, and

e) A cut-off point for end of month or end of year not clearly defined.

The performance of a system can also be affected by fluctuations in the volume of data. If there are not sufficient data collection staff to deal with peak loads a back-log builds up. When this is cleared by overtime or additional staff it can have disastrous effects on system performance.

3.5.1.2 *Changes in data preparation procedures* — Changes in these procedures are likely to have a significant effect on system performance. A cut in verification rate can result in more errors which means more machine time on error printing plus (usually) a second input when corrected. Also, inadequate training of new staff can cause problems if, for example, a punch operator is not aware of the installation standard for writing the letter '0' and the figure zero. Changes in input media from, say, punched cards to disk can cause problems if not handled carefully.

3.5.1.3 *On-line data entry* — There needs to be adequate control over personnel permitted access to the terminal. It is particularly important that new personnel should be properly trained in the correct way to enter data, and existing personnel are retrained when equipment is changed. It is essential that people who have the facility to interact with an on-line system are sufficiently familiar with the system and what it has been designed to do so that the implications of human intervention are thoroughly understood.

3.5.1.4 *Data validation* — Inadequate checking and cross checking of data being input to a system can result in all manner of problems that may affect the performance of the system in unpredictable ways.

Very few systems can afford to carry out total validation on all input data. All systems should, however, check that data are valid within reasonable limits. What is reasonable can change, however, as a result of modifications to the system and erroneous data can remain undetected by the existing validation routines.

3.5.1.5 *Parameter data* — Parameter data can be used for a variety of purposes including:

a) determining whether a programme functions in a weekly or a monthly mode,

b) establishing data validation limits,

c) setting constants or literals, and

d) determining the extent and nature of processing to be undertaken by general utility programmes.

The very nature of parameter data makes it difficult to validate on input because data that are within allowable limits can be incorrect for the intended purpose.

Where parameters are used to determine the extent and nature of processing by generalized utilities (for example, report generators), the run time may vary greatly depending on the parameters used. Misuse of such facilities can result in severe impact on other system areas and therefore adequate parameter validation and control over these runs is required. It is recommended that arrangements should be made for the data to be independently checked from a different source whenever possible.

Changes in staff, inadequate training of staff responsible for entering parameters and out-of-date instruction manuals are likely causes of error.

3.5.1.6 *Data brought forward* — Data that are brought forward may come from a previous run of the system that is being monitored or from another system. It is necessary to ensure that the system being monitored checks that the correct data have been brought forward. If the data brought forward have come from another system, checks should be made on the validity of the data by the system being monitored. Likely causes of performance degradation include:

a) amendments to the programmes or the system that wrote the data,

b) operational problems such as malfunction of the device on which the data were written, and

c) an additional unscheduled run of the system that wrote the data.

3.5.1.7 *Real time data* — Real time data are data that are intended to represent the immediate measured state of the environment, for example, data from sensors, analogue to digital converters, instrumentation and control equipment, radar systems.

In many cases the data source can itself be a complex related system with its own error and integrity checks so that main concern of the system being monitored will be the data transmission errors and possible periodic message integrity checks (for example, byte counts).

When the control of the data source comes within the province of the system being monitored, then in addition to the data transmission and message integrity checks, the system should include periodic integrity/calibration checks or runs on the data source.

It should be emphasized that, however, consistent the data appear to be to any integrity tests, if the data source is inconsistent with the environment the information content of the data will be impaired and the system performance will be degraded accordingly.

3.5.1.8 *Transmission channels* — Although operation of on-line systems should be transparent to users, various levels of error checking and retransmission will be occurring within each system. An increase in the number of retransmissions will occur if the transmission path becomes 'noisy' or unreliable, giving rise to increasing delays in the data transmission. This aspect should be monitored and warnings generated when the degradation reaches a predefined limit.

It should also be noted that changing the physical nature of a transmission channel can affect not only the bandwidth but also the type of noise or interference on the channel (for example, public/private, wires/ratio links, wire/optical fibre) and the error checking/monitoring routines may have to reflect this in order to minimize system degradation.

3.6 System Usage

3.6.0 *General* — Changes in the usage of the system being monitored can be reflected in modifications to the significant characteristics of the system inputs, processing or outputs or any combination of these. In some instances, the impact of a change will be readily apparent, for example, doubling the number of employees paid by the system will increase the size of the payroll run. In other instances the effect cannot be so easily predicted, for example, a change in an organization's sales discount structure may result in customers changing their ordering habits, thus affecting the sales invoicing routines.

Prominent among the aspects of system operation likely to be affected by changes in usage are:

a) increases or decreases in volumes of data,

b) revision of timing,

c) revision of quality controls, and

d) misuse of the system.

3.6.1 *Elements of Change*

3.6.1.1 *Increases or decreases in volumes of data* — Increases in volumes of data can be expected to pose a potential threat to system performance while decreases will normally be expected to have either a beneficial effect or none at all. Increases in volume can occur in, for instance, numbers of documents, numbers of items in a document,

numbers of items on a file, numbers of characters per data item, numbers and ranges of parameters.

It is advisable to trace the possible effect of any change through the sequence of input, processing and output to ensure that all possible effects on the performance of the system are fully explored.

3.6.1.2 *Revision of timing* — Revisions of timing can include changes to deadlines for inputs or outputs and frequencies of computer runs. These changes can threaten maintenance of schedules by exerting pressure on any or all of the inputs, processing, storage or output facilities available at the time.

3.6.1.3 *Revision of quality controls* — System performance can be adversely affected by revisions to the nature of the controls applied, including validation techniques, security dumps, access controls, boundary limits, reconfiguration rules. Again, it is advisable to trace the impact of change both forward and backward through the input, processing and output sequences to determine what effect they may have on the overall performance of the system.

3.6.1.4 *Misuse of the system* — A system can be run in a way not intended by the designer in order to obtain urgently required information. Although such actions can appear to have been achieved quite successfully, they can have serious and unforeseen secondry effects on, for example, data integrity, run frequencies, dependent systems and security.

3.7 Effects of Other Systems

3.7.1 *General* — For the purposes of this code, other systems are considered under two main categories:

a) Unrelated systems, and

b) Related systems.

An unrelated system is one that has no direct interface to the system being monitored but that shares resources with it. A change in use of the shared resources can therefore affect the performance of any system that uses them.

A related system is one that has a direct interface with the system being monitored. A change in an interface or in use of the related system can therefore affected the performance of the system being monitored. A related system can also include the characteristics of an unrelated system, that is, share resources.

3.7.2 *Elements of Change*

3.7.2.1 *Unrelated systems* — The major items that should be considered in these systems are those resources that can be shared with the system being monitored, as follows:

a) *Processor* — Any variation in the programmes that concurrently use the processor capacity can have an effect on performance, in particular the elapsed time and through-put. The variation may be the introduction of a new programme, which increases the overall load, or a change in priorities and scheduling, which results in a different mix of jobs. Increased usage of an existing programme or new arrangements for the use of the processor to act as standby or backup can also affect the overall loading and performance of the processor.

b) *Communications systems* — An increased load on communication lines and controllers by one system can affect the performance of another system that shares the communication facilities. A change in the configuration of the communications aspects of a system should be considered as a change in the hardware of that system and be reviewed under that heading. However, a communications configuration change, for example, new connections or new line speeds, can affect the performance of the mainframe processing system and therefore affect the performance of other systems, even if they have no communications capability.

c) *File store and peripherals* — A change in the size of usage of filestore can affect the performance of all filestore traffic and have a significant effect on any associated system. Re-allocation of filestore to different peripheral types can improve or degrade the performance of the system being monitored even though it does not use the changed filestore. Reconfiguration of peripherals or controllers that are outside the system being monitored can cause changes in the priorities or service requests within an operating system and hence change overall performance.

d) *Storage media* — The most significant changes arise when the media that form part of the system are directly changed. However, a change in the size or organization of a media library or handling mechanism can cause a change in the service provided. An increase in the media volumes, for example, packs, cartridges or reals, may also require a change in housekeeping and archiving practices which could affect the users of the volumes.

e) *Data preparation* — A change in data preparation equipment, the number of staff employed or the workload can affect any user of the service and therefore can affect the total behaviour of all systems that rely on that service.

f) *Operations* — A change in the total hardware configuration or the total workload or the number of operations staff can affect any system that is the responsibility of one group of operations staff.

3.7.2.2 *Related systems* — In addition to the major items listed for unrelated systems, the items to be considered in these system are those areas where there may be direct interfaces with the system being monitored:

a) *Interface data formats* — Any change in the format of data, even if not used by the system being monitored, can affect performance, for example, increased record length or new record types. Data to be considered can be transferred in processor main store, through filestore, on magnetic media, over communications lines or on paper. Change can be planned or accidental;

b) *Data sets* — A change in frequency of operation can introduce more data volumes to hold the same amount of data and there can be an indirect effect on performance;

c) *Procedures* — A change in operational procedures for one system can change the procedures for another; and

d) *New or changed system* — A new use of existing data can influence the old system to which it is related, for example, data rejection causing a requirement for more stringent vetting of the original input data.

4. MONITORING TECHNIQUES

4.0 General

4.0.1 This section identifies various types of monitoring techniques and provides guidance on the purposes for which each is best suited. Information is provided on each technique and what it achieves. General information and guidance is also included on the implications of the use of each technique and of the types of resources which may be required.

4.0.2 Each system to be monitored and its environment are unique and will require a different combination of monitoring techniques in order to be cost effective. In the event that a change to a critical quality is revealed by the chosen monitor, the user will need to investigate further, possibly with selective use of other monitoring techniques, in

order to discover which of the possible areas of change has actually caused the deterioration in performance.

4.0.3 The prospective user should review his system and the total operational environment before introducing any monitoring techniques and some of the questions that should be considered are as follows:

a) Which are the critical qualities which require monitoring ?

b) What is the cost (man and machine resources) of obtaining the data ?

c) What is the effect on the performance of the system of introducing a monitoring technique ?

d) What is the output from the monitoring technique ?

e) What training is required in order to introduce and implement a monitoring technique and process its output ?

4.1 Use of Routine Records

4.1.1 Records should be kept as a routine management control of the day to day running of a computer installation. The information contained in these records will very often provide the first indication of a charge in the performance of the hardware, software and human components of a computer-based system and, thence, of the system as a whole.

4.1.2 Classes of information that may be found useful include:

a) volumes of data handled;

b) system response times;

c) job run frequencies

d) achievements against run schedules;

e) equipment availability, utilization and idle time;

f) backlogs of work;

g) environment (temperature, humidity, etc);

h) errors in inputs, processing and outputs;

j) nature, frequency and duration of hardware faults;

k) nature and effect of software faults;

m) frequency and duration of power supply failures;

n) nature, frequency and duration of communication faults and their effects; and

p) change-control records.

4.1.3 Routine reports generated from these records should enable changes in performance to be detected. This may then lead to the application of specific monitoring techniques to identify the causes of these changes.

4.2 Hardware Monitors

4.2.0 *General* — A hardware monitor provides the most powerful performance measurement capability available to assist in the monitoring of computer systems. In simple terms a hardware monitor is an electronic device that is physically connected to the computer and that measures and record its activity. It is called a hardware monitor, not because it monitors the hardware, but because it is a piece of hardware as opposed to an item of software.

Whatever measurements might be taken by a hardware monitor during a performance analysis study, they are rarely sufficient in themselves to determine the causes of good or bad performance. The results of the measured utilization should be investigated further.

4.2.1 *Capabilities* — A hardware monitor is normally able to:

a) time or count the activity of every device (CPU, channels and peripherals) in a computer configuration;

b) distinguish between classes of activity within a device, for example, the component parts of a disk access (seek, latency, transfer, feed-back check);

c) measure the frequency of obeying identified instructions in a programme or the amount of time accessing each track of a direct access device;

d) analyse and display the results of measurements in real time, using in-built minicomputers and VDUs; and

e) impose no interference or degradation to the system hardware.

4.2.2 *When to Use a Hardware Monitor* — A hardware monitor can be usefully employed:

a) when the system hardware or software does not provide facilities suitable for a software monitor,

b) when a software monitor compatible with the system hardware is not available,

c) when the overheads imposed by a software monitor are not acceptable, and

d) when measurements beyond the abilities of a software monitor are required.

4.3 Software Monitors

4.3.0 *General* — A software monitor is a computer programme that resides in a computer system, the task of which is to collect information about the way the system is being used over a period of time. There are three categories of software monitor: programme monitors, system monitors and operating system data logging.

All three categories use system resources, the extent of which should be considered when interpreting results. It is also desirable to incorporate run time switches to permit them to be initiated, suppressed and used selectively.

4.3.1 *Programme Monitors*

4.3.1.0 *General* — Programme monitors report on the CPU utilization of an individual programme. If it is intended to extract a little more capacity from a configuration then it may well be worth considering a performance evaluation exercise concentrating on the five to ten most timeconsuming programmes. Programme monitors can be obtained commercially for a relatively low sum, but alternatively they can often be provided by the individual programmer/installation. There are three types of programme monitor.

4.3.1.1 *Programme instruction* — This instruction can count the number of times a routine is entered during a programme.

4.3.1.2 *A 'harness' monitor* — This monitor comprises a programme compiled into the subject programme providing information in such forms as histograms of the frequency with which any instruction is obeyed or traces of branching paths.

4.3.1.3 *Measurement by timing each instruction or routine* — This measurement can be done using either of the following methods:

a) *Interrogation of the central processor time facility by user programme* — This can impose severe overheads on the running time of the programme.

b) *Activity sampling* — This involves the random capture of the current contents of the instruction counter and can either be done with proprietary software or by using the central processor timer interrupt facility to activate the monitor at regular intervals. Currently available software has the advantage of in-built techniques to ensure the random nature of the sample.

4.3.2 *System Monitors*

4.3.2.0 *General* — System monitors collect information on the performance of the system as a whole including channel and peripheral

utilization. They are essentially packages working on a sampling basis similar to the programme monitors described in 4.3.1 but capturing copies of tables and lists maintained by the operating system showing the status of the system at any one time. However, they demand resources from the system they are monitoring; amounts vary but are likely to be significant. The copies produced are stored and subsequently analysed by a second package which produces a comprehensive series of reports detailing the utilization levels of the various hardware components, including store utilization and central processor usage, broken down between operating systems and user programme. It is usual to obtain system software monitors from independent software houses by purchase or lease.

4.3.2.1 *Capabilities* — A system monitor is:

a) simple to use; and

b) capable of producing output records relating to the logical components of the system (for example, usage of system files is reported under file name rather than the physical device concerned).

4.3.3 *Operating System Data Logging*

4.3.3.0 *General* — This facility can be considered as a software monitor and the value of the information provided to the performance measurement process can be enormous.

4.3.3.1 *Capabilities* — The data can:

a) show the hardware resources required to process a given work-load;

b) provide information for the diagnosis of causes of variations in system performance, for example, excessively high demand for particular system resources during certain phases of the work-load will explain degraded processing times; and

c) provide information as a basis for predicting levels of performance.

4.4 In Service Testing

4.4.0 *General* — During acceptance testing the user should have ensured that the system will meet his requirements. He will also need to repeat certain tests to ensure that the system continues to meet his performance requirements. This task can be divided into:

a) ensuring that the hardware continues to function correctly; and

b) ensuring that the system as a whole continues to provide satisfactory service.

Both categories usually involve running bench mark programmes where the expected result has been predetermined. It is not necessary to carry out regular tests on the software as it does not degrade with use or time. Any changes to the software should be subject to the normal acceptance test procedures.

4.4.1 *The System Hardware* — It is important to remember that system hardware in this context includes power lines, voltage stabilizers, no-break systems and similar equipment. The bench marks for in-service testing of all the major hardware elements are usually provided by the manufacturer, who should also provide diagnostic facilities. The checks need to be performed periodically for preventive maintenance purposes. In addition, certain elements of the hardware may need to be monitored continuously where they are critical to performance.

4.4.2 *The Total System* — Testing of the performance of the complete system involves constructing some form of bench mark test where the bench mark bears some known and close relationship to the usual work profile. The user will need to consider which method to use to apply the bench mark, for example, externally through a peripheral device or internally using a data base. Also, it is necessary to consider the method to be used to measure the relevant characteristics. It should also be appreciated that, although greater accuracy may be achieved when the bench mark has a close relationship to the actual environment, the cost factor can be significant and a more loosely coupled relationship may give a more cost effective indication.

4.4.2.1 As the characteristics of the system will vary with the work load it is important to establish the effect of high load conditions on through-put, response time, and availability at different levels of load. It will also show how different failure modes will occur.

4.4.2.2 Suitable bench marks can be used to show that corrective action taken to fix errors has been effective. In a similar way bench marks can be used to optimize a performance of a system by trading off one characteristic against another. This should be done by studying the effects of various parameters under constant bench mark test and then seeing the result of making discrete changes, that is, introducing high speed memory or changing queuing arrangements to remove bottlenecks.

4.5 System Audit

4.5.0 *General* — One of the objectives of a system and it is to check for changes in the performance characteristics of the system. These changes can indicate a degradation of the system or alternatively may be required to show an improvement following some earlier corrective action. To audit a computer-based system it is necessary to perform

some checks both manually and by automatic means and then to compare the results with historical records. In an ideal situation, special bench mark programmes can be run whose results would give an indication of performance changes since they were last executed, these bench mark programmes having been produced as part of the original system and acceptance tests.

In order to be able to define precisely the reasons for a particular change in parameter, the audits need to be comprehensive.

4.5.1 *Utilization Time by Users* — This information will show which areas of the organization are increasing their use of the system and those that are in decline. In a teleprocessing system this information can suggest that a redistribution to access nodes may be necessary.

4.5.2 *Mean Number of Jobs Processed* — This analysis will indicate the type of programmes that are used most frequently and those which should be most efficient when running.

4.5.3 *Mean Time to Process Jobs* — This characteristic will show how long jobs have taken to run; this will include any queuing time. It may be necessary to adjust a priority structure to the system to achieve a satisfactory balance or to increase the speed of some common peripherals.

4.5.4 *Mean Time to Prepare Jobs* — This time will indicate how well the manual part of the system is functioning. If long delays are occurring then additional staff or more efficient facilities may be required.

4.5.5 *Time Taken to Archive* — If this time shows excessive increase then more effective means should be found as the integrity of the system could be at risk.

4.5.6 *Unsuccessful Attempts to Acquire Peripherals* — Use should be made of any in-built fault logs; these logs should record the number of times the operating system has been unsuccessful in acquiring a peripheral. This number could indicate excessive loading or some faulty equipment.

4.5.7 *Spare Storage* — The rate of consumption of spare storage since the last audit should be considered. If excessive, this consumption rate could indicate either the need to provide additional capacity, or a poor allocation/collecting algorithm within the system.

4.5.8 *Amount of Useful Spare Processor Time Remaining* — This parameter should be assessed accurately as it may limit the efficiency of the system. If it is inadequate and process completion times are rising, then additional capacity should be procured or more economic programmes produced.

4.5.9 *System Down Time Since Last Audit* — The audit needs to consider the amount of down time since the last audit and attempt to determine any systematic pattern.

4.5.10 *Discrepancy Reports and Change Notes* — In addition to examining the data related to performance characteristics, the audit should consider discrepancies notified by the user and associated corrective action. These considerations require that the user organization has set up procedures for dealing with:

a) discrepancy reporting, and

b) change control.

The audit should try to correlate the perturbation in performance characteristics with deficiency reports. If correlation cannot be established for particular deficiencies, the auditor should consider if additional characteristics need to be measured. When planning an audit, consideration should be given to changes implemented in response to deficiency reports to determine what effect this action had on the performance of the system.

4.6 User Surveys

4.6.0 *General* — A good system is one that meets the requirements of the user, to the user's satisfaction. The techniques recommended in this standard can be used to ensure that the stated requirements of the user continue to be met. However, as user awareness of the system and computer systems in general increases, expectations may change and dissatisfaction set in. A dissatisfied user might disregard the system, use it carelessly or misuse it and the effect will be a detrimental change in system performance.

4.6.0.1 The user survey, as a technique of approaching users, can be compared to an opinion poll. The questions should be carefully constructed so as to be precise and not misleading. The compiler of the survey should take care that the questions themselves do not lead to dissatisfaction by suggesting the system could be bettered. The questions should also be within the scope and comprehension of the user.

4.6.0.2 The methods by which the survey may be conducted should be considered. A regular questionnaire, distributed by post/internal memo, to be filled in and returned, will provide bare facts on a regular basis, but not all users will respond. An experienced interviewer may get a better response and be able to assess the users' answers but this is very time consuming.

4.6.1 *Collection of Facts* — A user survey can be used to discover changes in requirements and the effect of changes made both to the

system being monitored and other systems which may affect it, for example:

a) volume of input,

b) frequency of use,

c) increased work load caused by a change to some other system, and

d) response time.

4.6.2 *Determination of the Level of Satisfaction* — Although the performance of a system may change, it is also necessary to determine the importance of the change to the user. A system may run unchanged in all respects other than the level of expectation of the user, or with continual use he may come across operating difficulties which will cause dissatisfaction or annoyance, for example:

a) the 'abort' button is next to 'accept' button on the keyboard,

b) he has seen a different design of VDU, and

c) the space on a report is not sufficient for additional handwritten notes.

4.6.3 *Prevention of Dissatisfaction* — The user survey may prevent psychological dissatisfaction growing, as it is evidence of the designer's continuing concern for the user and that the user is the driving force behind the system not *vice-versa.*

5. PERFORMANCE MONITORING REPORTING

5.0 General

5.0.1 This section makes recommendations to assist the user to establish a system of performance monitoring reporting that enables changes in performance detected by monitoring techniques to be analysed and presented in a meaningful fashion.

5.0.2 When deciding upon the scope of a reporting system it is necessary to be clear about its purpose and value so that true cost effectiveness is achieved. Inadequate reporting can fail to fulfil its purpose while a superfluity of records and reports can be both costly and lead to important trends being overlooked among a mass of valueless information.

5.0.3 It is important that management specifies clearly the form in which performance information should be presented.

5.1 Performance Criteria

5.1.0 *General* — An essential foundation of performance monitoring

reporting is a schedule of the criteria by which the system to be monitored will be assessed, together with an acceptable value (or range of values) for each criterion. Because system development and operation is often a dynamic process, this schedule will contain a mixture of original criteria and revised and additional criteria.

5.1.1 *Original Criteria* — These criteria should be obtainable from the system specification described in IS : 11290-1985* and should be set out in the form of a schedule of criteria to be met by the system being monitored.

5.1.2 *Revised Criteria* — Revised criteria can arise from:

a) amendments made during system development and documented in accordance with relevant standard (IS : 11290-1985*).

b) temporary or permanent concessions recorded on the test summary described in relevant standard (IS : 11291-1985†).

c) changes made subsequent to the handover of the system.

5.2 Evaluation of Performance Measurement Data

5.2.1 From the data produced by the chosen monitoring techniques, measures of achievement can be recorded for comparison with the scheduled performance criteria. In assessing the significance of these comparisons it is important to take account of:

a) any assumptions made with respect to the conditions in which the monitoring took place,

b) problems encountered in applying the monitoring techniques, and

c) how accurately the techniques applied actually measure the criteria being monitored.

5.2.2 It is also important to ensure that all significant measures and comparisons are truly comparing like with like.

Similarly careful consideration should be given to the period of time or number of occurrences which should be included in these comparisons and records so that, for example, periodic variations or long term trends do not escape attention.

5.3 Performance Reporting

5.3.1 Care should be taken in determining the precise form in which performance reports should be presented so that their practical or

*Code of practice for documentation of computer based system.

†Code of practice for testing of computer based system.

economic significance is made perfectly clear to the recipients. The nature of the contents may suggest whether they should take the form of, for example:

a) exception reports;

b) statistical analyses and summaries; and

c) trend reports (for example, charts or graphs).

5.3.2 In additional to the form of presentation, the time at which reports are submitted (in relation to the comparisons which they contain) can be of importance for management control. Consideration should be given to the cost effectiveness of having reports made:

a) immediately a change in performance is detected;

b) on trends reaching predetermined levels or values; and

c) periodically (for example, daily, weekly, monthly, annually).

5.3.3 Consideration should also be given to the possibility that the application of alternative reporting criteria could assist management decision making. For example, a system loading report may be made at set periods but, in addition, immediately a load exceeding a given value is detected.

5.3.4 Because computer-based systems tend to be dynamic in nature, it is essential to ensure that the monitoring and reporting is reviewed on a regular basis to ensure that it is at all times:

a) relevant to the current environment,

b) presented in the most suitable form,

c) timely in its submission, and

d) cost justifiable.

INDIAN STANDARDS INSTITUTION

Headquarters:

Manak Bhavan, 9 Bahadur Shah Zafar Marg, NEW DELHI 110002

Telephones : 3 31 01 31, 3 31 13 75 | Telegrams: Manaksanstha (Common to all offices)

Regional Offices:	*Telephone*
*Western : Manakalaya, E9 MIDC, Marol, Andheri (East), BOMBAY 400093	6 32 92 95
†Eastern : 1/14 C.I.T. Scheme VII M, V. I. P. Road, Maniktola, CALCUTTA 700054	36 24 99
Southern : C. I. T. Campus, MADRAS 600113	41 24 42
Northern : B69 Phase VII, Industrial Focal Point, S. A. S. NAGAR 160051 (Punjab)	8 73 28

Branch Offices:

'Pushpak', Nurmohamed Shaikh Marg, Khanpur, AHMADABAD 380001	2 63 48 2 63 49
'F' Block, Unity Bldg, Narasimharaja Square, BANGALORE 560002	22 48 05
Gangotri Complex, Bhadbhada Road, T. T. Nagar, BHOPAL 462003	6 27 16
22E Kalpana Area, BHUBANESHWAR 751014	5 36 27
5-8-56C L. N. Gupta Marg, HYDERABAD 500001	22 10 83
R14 Yudhister Marg, C Scheme, JAIPUR 302005	6 98 32
112/418 B Sarvodaya Nagar, KANPUR 208005	4 72 92
Patliputra Industrial Estate, PATNA 800013	6 23 05
Hantex Bldg (2nd Floor), Railway Station Road, TRIVANDRUM 695001	32 27

Inspection Office (With Sale Point):

Institution of Engineers (India) Building, 1332 Shivaji Nagar, PUNE 411005	5 24 35

*Sales Office in Bombay is at Novelty Chambers, Grant Road, Bombay 400007 — 89 65 28

†Sales Office in Calcutta is at 5 Chowringhee Approach, P. O. Princep Street, Calcutta 700072 — 27 68 00

Printed at Swatantra Bharat Press, Delhi (India)